UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/963,547	09/27/2001	Gabi Malka	2207/10125	9253

23838 7590 10/07/2003

KENYON & KENYON
1500 K STREET, N.W., SUITE 700
WASHINGTON, DC 20005

EXAMINER	
CHEN, PO WEI	
ART UNIT	**PAPER NUMBER**
2672	6

DATE MAILED: 10/07/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/963,547	MALKA ET AL.	
	Examiner	Art Unit	
	Po-Wei (Dennis) Chen	2676	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *30 July 2003* .

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-17* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-17* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)☐ The proposed drawing correction filed on _____ is: a)☐ approved b)☐ disapproved by the Examiner.

If approved, corrected drawings are required in reply to this Office action.

12)☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____ .

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

a) ☐ The translation of the foreign language provisional application has been received.

15)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ .
4) ☐ Interview Summary (PTO-413) Paper No(s). _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: .

DETAILED ACTION

In response to an Amendment received on July 30, 2003. This action is final.

Claims 1-17 are pending in this application. Claims 1, 6, 9, 11 and 16 are independent claims.

The present title of the invention is "Texture Engine State Variable synchronizer".

The Group Art Unit of the Examiner case is now 2676. Please use the proper Art Unit number to help us serve you better.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1, 4, 6-9 and 16-17 are rejected under 35 U.S.C. 103(a) as being unpatentable by Migdal et al. (US 6,392,655; refer to as Migdal herein) and further in view of Duluk, Jr. et al (US 6,525,737; refer to as Duluk herein).

3. Regarding claim 1, Migdal discloses a method for multiple texture rendering comprising:

A method for synchronizing texture pipelines in a graphics engine ("Recirculating texture blender 208 accepts four textures numbers 212 and four multiple filtered texture values 216 output from texture filter 207", see lines 37-51 of column 10 and Fig. 1, 2 and 5). It is noted that while claim recites synchronizing pipelines, which represents a sequence of processes, it is clear that in Fig. 5, there are four textures are processed in separated sequences depending on the texture number and values received. Therefore, each texture process can be considered as a pipeline that is synchronized by the texture number and values. Also, while claim recites engine,

it is clear that the raster subsystem (element 124 of Fig. 1) functions the same as a graphic engine. Thus, limitation of claim is met;

Loading polygon state variables into a plurality of sets of texture pipeline state variable queues ("Hardware 302 receives texture number 212 from fine grain scan converter 204 and performs texture processing using both texture number 212 and the state information corresponding to texture number 212 that is stored in one of state registers 304" and "Scan converter 410 takes in a primitive description 210...texture coordinate calculator 440 then outputs texnum 212 and the calculated texture coordinates 214", see lines 13-35 of column 9 and lines 59-67 of column 9 and lines 1-22 of column 10 and Fig. 4-5). It is noted that while claim recites polygon state variables, it is clear that by loading primitive (it is well known in the art that the term primitive is used instead of polygon) description, the system produces a texture number that functions the same as a state variable for the primitive inputted. The texture number is then inputted to one of the texture state registers which stores states information for different textures.

Enabling a texture processing portion of a number of the sets of state variable queues corresponding to a number of parallel texture operations indicated by the polygon state variables ("Corresponding to each of the texture state registers 304A-n is an enablement register indicate whether a particular texture is enable 306A, 306B, 306C,...306n", see lines 22-25 of column 9 and lines 61-67 of column 11 and Fig. 5; also see lines 9-12 of column 3).

It is noted that Migdal does not disclose an accumulation portion for the state variables and sets of parallel texture pipeline state variable queues. However, this is known in the art taught by Duluk. Duluk teaches a graphic processor with pipeline state storage and retrieval that "state changes are accumulated until the spatial information for a primitive (i.e., the completing

vertex) is received, and those accumulated states are in effect during the rendering of that primitive" and "As a high-performance alternative, multiple pipelines are run in parallel" (see lines 20-29 of column 4 and lines 14-15 of column 17).

It would have been obvious to one of ordinary skill in the art at the time of invention to utilize the teaching of Duluk to provide the advantage of producing image efficiently with a high performance process (see lines 58-60 of column 2 and lines 14-15 of column 17, Duluk). Also, both Migdal and Duluk are directed to a graphic processing utilizing state parameters on rendering polygons, or primitives.

4. Regarding claim 4, Migdal discloses a method for multiple texture rendering comprising:

Disabling the texture processing portions of the remaining sets of state variable queues ("Texturing is enabled and disabled individually for each texture. If texturing is disabled for one of the textures, then the fragment result from the previous stage is passed unaltered to the next stage", see lines 5-8 of column 17. Also see 13-25 of column 9).

5. Regarding claim 6, Migdal discloses a method for multiple texture rendering comprising:

A method of synchronizing multiple texture pipelines ("Recirculating texture blender 208 accepts four textures numbers 212 and four multiple filtered texture values 216 output from texture filter 207", see lines 37-51 of column 10 and Fig. 1, 2 and 5). It is noted that while claim recites pipelines that represents a sequence of processes, it is clear that in Fig. 5, there are four textures are processed in separated sequences depending on the texture number and values received. Therefore, each texture process can be considered as a pipeline. Thus, limitation of claim is met;

For a predetermined number N of the texture pipelines, advancing the state variable data to succeeding portions of the texture pipeline, N representing a number of textures to be applied to polygon data (see lines 13-35 of column 9 and Fig. 3 and 5 and lines 44-58 of column 11). It is noted that depending on the number of textures are being processed on a primitive, or polygon, different number of texture pipelines are enabled to process using the state variable data in the state variable registers. Thus, limitation of claim is met.

It is noted that Midgal does not disclose accumulating state variable data and multiple parallel pipelines. However, this is known in the art taught by Duluk, as statements presented above, with respect to claim 1 are incorporated herein.

6. Regarding claim 7, with statements presented above, with respect to claim 4 are incorporated herein.

7. Regarding claim 8, it is noted that Migdal does not disclose receiving new state variable data, the new state variable data being defined differentially with respect to old state variable data previously accumulated, and evicting obsolete elements of the old state variable data in favor of the new state variable data. However, this is known in the art taught by Duluk. Duluk teaches a graphic processor with pipeline state storage and retrieval comprising "the value of a state parameter remains in effect until it is changed, and changes simply overwrite the older value because they are no longer needed" (see lines 14-36 of column 4). It would have been obvious to one of ordinary skill in the art at the time of invention to utilize the teaching of Duluk to provide the advantage of producing image efficiently (see lines 58-60 of column 2, Duluk). Also, both Migdal and Duluk are directed to a graphic processing utilizing state parameters on rendering polygons, or primitives.

8. Regarding claim 9, Migdal discloses a method for multiple texture rendering comprising:

A control method for a texture processing system having multiple texture pipelines (see lines 37-51 of column 10 and Fig. 1, 2 and 5);

If the texture processing system switches modes, transitioning from a first number of active texture pipelines to a second number of active texture pipelines; and then, in each of a number of texture pipelines corresponding to the second number, advancing the state variable data from the respective registers to a remainder of the respective state variable queues, and disabling the remaining texture pipelines and portions of associated state variable queues (see lines 13-35 of column 9, lines 44-58 of column 11 and lines 36-51 of column 10 and Fig. 3 and 5). It is noted that depending on the number of textures are being processed on a primitive, or polygon, different number of texture pipelines are enabled or disabled to process using the state variable data in the state variable registers. Thus, limitation of claim is met.

It is noted that Migdal does not disclose accumulating state variable data in a register, the state variable data being received over a plurality of clock cycles as a plurality of data units and multiple parallel pipelines. However, this is known in the art taught by Duluk, as statements presented above, with respect to claim 1 above, are incorporated herein. Also see lines 28-33 of column 9. It is noted that MEX accumulates state information using a register. While claim recites a plurality of clock cycles, it would have been obvious to one of ordinary skill in the art at the time of invention to understand that any computer process is timing in terms of clock cycles. In a graphic processing pipeline system such as one disclosed by Duluk, performing different process such as accumulating state information in a pipeline will go through a plurality of clock cycles. Thus, limitation of claim is met.

9. Regarding claim 16, Migdal discloses a method for multiple texture rendering comprising:

A method of synchronizing texture processor (lines 37-51 of column 10 and lines 8-13 of column 5 and Fig. 1, 2 and 5);

A plurality of state variable queues (see lines 13-35 of column 9 and lines 44-58 of column 11 and Fig. 3 and 5).

Each of said state variable queues associated with a texture pipeline processor (lines 13-23 of column 3 and Fig. 3 and 5);

Advancing the polygon state variables from the respective portion to a texture processing portion of the texture pipeline processor (see lines 13-35 of column 9, lines 44-58 of column 11 and lines 36-51 of column 10 and Fig. 3 and 5).

Advancing performed only for a number of portions corresponding to a number of parallel texture operations indicated by the polygon state variables (see lines 13-35 of column 9, lines 44-58 of column 11 and lines 36-51 of column 10 and Fig. 3 and 5). It is noted that depending on the number of textures are being processed on a primitive, or polygon, different number of texture pipelines are enabled to process using the state variable data in the state variable registers. Thus, limitation of claim is met.

Migdal does not disclose an accumulation portion for the state variables and parallel texture pipelines processor. However, this is known in the art taught by Duluk, with statements presented above, with respect to claim 1 above are incorporated herein. Also see lines 55-61 of column 19 of Duluk.

10. Regarding claim 17, Migdal discloses a method for multiple texture rendering comprising:

Enabling the texture processing portions that receive the advanced polygon state variables (see lines 13-35 of column 9, lines 44-58 of column 11 and lines 36-51 of column 10 and Fig. 3 and 5);

Disabling the remaining texture processing portions (see lines 13-25 of column 9 and lines 5-8 of column 17).

Migdal does not disclose an accumulated polygon variables. However, this is known in the art taught by Duluk, with statements presented above, with respect to claim 1 above are incorporated herein.

11. Claims 2-3 are rejected under 35 U.S.C. 103(a) as being unpatentable over Migdal (US 6,392,655; refer to as Migdal herein) and Duluk, Jr. et al (US 6,525,737; refer to as Duluk herein) as applied to claim 1 above, and further in view of Fliflet (US 2002/0140710).

12. Regarding claim 2, Migdal discloses a method for multiple texture rendering comprising:

Receiving the polygon state variables in a state variable accumulator (see lines 28-35 of column 9). It is noted that each state register receives texture number which was generated by the inputted primitive description. Thus, texture number can be considered as polygon state variable which is stored in the state register when received. It is noted that Migdal does not disclose copying the received polygon state variables to a state variable latching register. However, this is known in the art taught by Fliflet. Fliflet discloses a method to balance software and hardware in a graphic rendering system that "determine if the fast state group has been written into a corresponding zone buffer. If not, the binning driver proceeds to write a copy

of the fast state group into the bin buffer" (see paragraph 0027 and 0031; while claim recites latching register, it is clear that the bin buffer, disclosed by Fliflet, will function the same to store the state variables). It would have been obvious to one of ordinary skill in the art at the time of invention to utilize the teaching of Fliflet to optimizing system performance by using hardware's cache functionality (see paragraph 0012, Fliflet). Also, both Migdal and Fliflet are directed to graphic rendering system.

13. Regarding claim 3, it is noted that Migdal does not disclose copying is performed prior to processing each polygon. However, this is known in the art taught by Fliflet, as statement presented above, with respect to claim 2 are incorporated herein. Further, it is noted that Fliflet disclose a method to improve the system performance by caching state variables in a bin buffer for primitives rendering process. Therefore, it is clear that the step of copying state variables into the buffer is performed prior to processing each primitive, or polygon. Thus, limitation of claim is met. It would have been obvious to one of ordinary skill in the art at the time of invention to utilize the teaching of Fliflet to optimizing system performance by using hardware's cache functionality (see paragraph 0012, Fliflet). Also, both Migdal and Fliflet are directed to graphic rendering system.

14. Claims 5 and 10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Migdal (US 6,392,655; refer to as Migdal herein) and Duluk, Jr. et al (US 6,525,737; refer to as Duluk herein) as applied to claims 1 and 9 above, and further in view of Melo et al. (US 6,243,817; refer to as Melo herein).

15. Regarding claim 5, Migdal discloses a method for multiple texture rendering comprising:

Disabled texture processing portions (see lines 5-8 of column 17. Also see 13-25 of column 9).

It is noted that Migdal does not disclose removing power for disabling. However, this is known in the art taught by Melo. Melo discloses a method for dynamically reducing power consumption that "selectively removing power to the second set of input buffers if the second set of signals are inactive" (see lines 36-41 of column 3). It would have been obvious to one of ordinary skill in the art to utilize the teaching of Melo to provide the advantage of reducing power consumption (see lines 24-26 of column 2, Melo). Also, both Migdal and Melo are directed to disable or inactivate certain component of a computer system.

16. Regarding claim 10, as statements presented above, with respect to claim 5 are incorporated herein.

17. Claims 11-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Migdal (US 6,392,655; refer to as Migdal herein) and further in view of Duluk, Jr. et al (US 6,525,737; refer to as Duluk herein) and Battle (US 6,462,743).

18. Regarding claim 11, Migdal discloses a method for multiple texture rendering comprising:

A texture processing system, comprising: a plurality of texture pipelines ("Recirculating texture blender 208 accepts four textures numbers 212 and four multiple filtered texture values 216 output from texture filter 207", see lines 37-51 of column 10 and Fig. 1, 2 and 5). It is noted that while claim recites pipelines, which represents a sequence of processes, it is clear that in Fig. 5, there are four textures are processed in separated sequences depending on the texture number and values received. Therefore, each texture process can be considered as a pipeline.

A plurality of state variable queues, at least one state variable queue provided for each texture pipeline, the state variable queues each coupled to a series of state variable queue processing stages (see lines 13-35 of column 9 and Fig. 3 and 5 and lines 44-58 of column 11). It is noted that depending on the number of textures are being processed on a primitive, or polygon, different number of texture pipelines are enabled to process using the state variable data in the state variable registers. Also, it is noted that while claim recites accumulating, by storing state variables over clock cycles does perform the same as accumulating. See lines 14-15 of column 3 and lines 1-7 of column 5. Thus, limitation of claim is met.

A controller having a first control output adapted to disable at least one of the series of state variable queue processing stages (see lines 62-67 of column 16 and lines 1-8 of column 17 and Fig. 5).

Midgal does not disclose an accumulation register and multiple parallel pipelines. However, this is known in the art taught by Duluk, as statements presented above, with respect to claim 9 are incorporated herein.

Midgal does not disclose a latching register. However, this is known in the art taught by Battle. Battle teaches a pipeline processing comprising "parameter registers 52 latch the values on parameter bus 44 when prompted to do so" (see lines 47-61 of column 5). It would have been obvious to one of ordinary skill in the art to utilize the teachng of Battle to provide the advantage of enhanced efficiency in a graphic pipeline processing system (see lines 21-26 of column 2, Battle). Also, both Midgal and Battle are directed to a graphic system that utilize state variables for rendering primitives. Thus, limitation of claim is met.

19. Regarding claim 12, Migdal discloses a method for multiple texture rendering comprising:

Controller has a data input adapted to receive a state variable from a programming source (see lines 64-67 of column 10 and Fig. 5).

20. Regarding claim 13, Migdal discloses a method for multiple texture rendering comprising:

Controller has a data output adapted to transfer a received state variable (see lines 64-67 of column 10 and Fig. 5). It is noted that state variable such as texture number (element 212 of Fig. 5) is received by the controller (element 510 of Fig. 5) and output the texture number to next stage of the pipeline process.

Migdal does not disclose an accumulation register. However, this is known in the art taught by Duluk, as statements presented above, with respect to claim 11 are incorporated herein.

21. Regarding claim 14, Migdal discloses a method for multiple texture rendering comprising:

Controller has a second control output adapted to trigger the transfer of state variables from the register to the corresponding register of each state variable queue (see lines 64-67 of column 10 and Fig. 5). It is noted that state variable such as texture number (element 212 of Fig. 5) is being triggered by controller (element 510 of Fig. 5) to go from a state register to the next state register of next stage (element 512) of the pipeline process.

Midgal does not disclose an accumulation register. However, this is known in the art taught by Duluk, as statements presented above, with respect to claim 11 are incorporated herein.

Midgal does not disclose a latching register. However, this is known in the art taught by Battle, as statements presented above, with respect to claim 11 are incorporated herein.

22. Regarding claim 15, Migdal discloses a method for multiple texture rendering comprising:

A computer system (see Fig. 1);

A processor coupled to a bus (see lines 31-33 of column 6 and element 101 and 102 of Fig. 1);

A system memory in communication with the bus (see lines 31-33 of column 6 and element 102 and 104 of Fig. 1);

A graphics processor comprising the texture processing system (see element 120 of Fig. 1).

Response to Arguments

23. Applicant's arguments filed July 30, 2003 have been fully considered but they are not persuasive.

The Applicant argues the combination of references Migdal and Duluk does not disclose the usage of parallel texture pipelines in independent claims 1, 6, 9, 11 and 16. However, this is known in the art taught by Duluk. Duluk teaches a graphic processor with pipeline state storage utilize the method (lines 14-20 of column 17). It would have been obvious to one of ordinary skill in the art at the time of invention to utilize the teaching of Duluk to provide the advantage of producing image efficiently with a high performance process (see lines 58-60 of column 2 and lines 14-15 of column 17, Duluk). Also, both Migdal and Duluk are directed to a graphic processing utilizing state parameters on rendering polygons, or primitives.

Conclusion

24. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Kwok et al. (US 6,088,044) disclose "Method for Parallelizing Software Graphics Geometry Pipeline Rendering".

Rosasco (US 6,317,137) disclose "Multi-Threaded Texture Modulation for Axis-Aligned Volume Rendering".

Denneau et al. (US 6,384,833) disclose "Method and Parallelizing Geometric Processing in a Graphics Rendering Pipeline".

Simha et al. (US 6,476,810) disclose "Method and Apparatus for Generating a Histogram of a Volume Data Set".

25. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Inquiry

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Po-Wei (Dennis) Chen whose telephone number is (703) 305-8365. The examiner can normally be reached on 9am-5pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matthew C Bella can be reached on (703) 308-6829. The fax phone number for the organization where this application or proceeding is assigned is (703) 872-9306.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

Po-Wei (Dennis) Chen
Examiner
Art Unit 2676

Po-Wei (Dennis) Chen
September 30, 2003

MATTHEW C. BELLA
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600